(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 661 620 A1**

(12) EUROPEAN PATENT APPLICATION

(21) Application number: **94309234.6**

(51) Int. Cl.6: **G06F 3/033**

(22) Date of filing: **09.12.94**

(30) Priority: **30.12.93 US 175853**

(43) Date of publication of application:
05.07.95 Bulletin 95/27

(84) Designated Contracting States:
DE FR GB

(71) Applicant: **XEROX CORPORATION**
Xerox Square
Rochester
New York 14644 (US)

(72) Inventor: **Moran, Thomas P.**
1037 Greenwood Avenue
Palo Alto, CA 94301 (US)
Inventor: **Chiu, Patrick**
564 University Drive No. 3
Menlo Park, CA 94025 (US)

(74) Representative: **Reynolds, Julian David et al**
Rank Xerox Ltd
Patent Department
Parkway
Marlow
Buckinghamshire SL7 1YL (GB)

(54) Apparatus and method for executing multiple concatenated command gestures in a gesture based input system.

(57) A graphical editing system is provided which allows a user to draw number of commonly used gestures (70-80) called primitive command gestures (Figs.4,6) together in essentially one stroke. The primitive gestures (70-80) are chosen to represent gestural syntax, analogous to a textual command syntax. The primitive gestures (70-80) are concatenated. i.e., connected in a series of commands by a gestural operator called a break feature (76,A). Every input composite gesture (Fig.7) is scanned to detect the break features (76,A). When found, the composite gesture is broken up into primitive gestures (70-80) and each gesture is interpreted and executed in the order in which the individual gestures were drawn. The preferred break feature is sharp angle (A) drawn between consecutive primitive gestures(70,80). In a composite gesture consisting of two primitive gestures, one being the selection gesture (70), the break featur is recogniz d as any departure from the selection gesture.

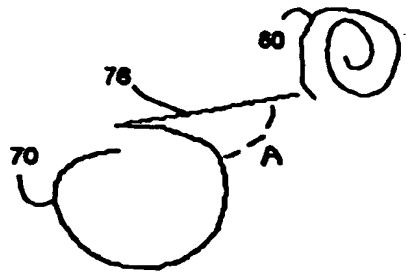

Fig 7

EP 0 661 620 A1

This invention pertains to an information input system and method and more particularly to a "gesture based" graphical input system that receives information by sensing the motion of a stylus, pen, finger or mouse.

Computer graphic operations that are initiated by pointing devices are typically two types: drawing operations and control operations. Drawing operations describe loci for the placement of visible marks on the image, analogous to drawing on a piece of paper by making a motion with a pen in contact with the surface. Control operations are used to initiate and guide the execution of computer functions leading to modification of the visible marks. Control operations are especially important to the performance of editing functions such as erasing, moving, or changing the visual qualities (e.g. line width) of visible marks of the image.

It is noted that the basic function of gesture-based input such as a pen-based notebook computer or the Liveboard is to allow the user to make freehand marks by simply drawing ("inking") them. Such a mark is then a data object in the system. Operations can then be performed on these objects by particular control functions known as gestures. A gesture is a hand-drawn command and it is a mark created with a stylus input device that is interpreted by the system as designating an action for the system to perform. In order to institute a simple control command by gesture, the user would draw a gesture which represents some desired function and then commit to the command by lifting the input device (e.g. stylus) from the input surface of a display or releasing a button on the device (e.g. mouse). That is, the completed gesture is sensed and after lifting the stylus from the input surface of the display the computer is instructed to execute the particular command.

In the "whiteboard" system for the Liveboard, gestures have been frequently used as the preferred procedure for issuing many commands on the input surface of the display. One example of a command is the selection gesture. There are many types of command gestures for selecting objects written by a user. The most common is a loop gesture in which objects are selected by drawing a circle around such objects by a freeform gestural stroke. Fig. 2 illustrates such a freeform loop selection at 32. A plurality of objects drawn on display screen 14 are selected for future operation by encircling such objects with freeform loop selection 32.

Once selected, other gestures can operate on the selected objects. In some instances, this previously described sequence of events, i.e., (1) selecting specified objects, (2) pausing between selection and subsequent operation, and (3) and executing subsequent operations on selected objects, is desirable because a user has the option of editing or modifying his/her selection. Th act of selecting is often times complex enough to require a user's complete attention. Once selected, the user can then perform other operations on the selected objects. In the preferred system, the selection is maintained after an operation so several operations can be performed until the user achieves what is intended.

In many other cases where the selection is simple, a user may desire to save time, i.e., eliminate the pause between selecting objects and subsequent operation by combining or concatenating two or more gestures in essentially one stroke. That is, the user may wish to draw a "composite" gesture, i.e., a number of separate individual gestures combined or connected in series to perform a number of desired functions in sequence, much like that procedure used with several text commands. In a text command environment, text commands are combined to create a command sequence in which a number of desired functions are performed to specified data. The textual commands connected in series are separated by punctuation, e.g., semi-colons. The punctuation is then detected to separate individual textual commands. Such commands are then interpreted and executed in the sequence drawn.

In a gesture based input system, it would also be desirable to concatenate a number of separate individual gestures in one stroke to perform a number of desired functions.

It is an object of the present invention to allow a user to combine a number of single stroke gestures into one compound single stroke gesture to perform.

The above object is achieved in accordance with the present invention by a a system which allows a user to draw number of gestures called primitive gestures together in essentially one stroke. The combination of primitive gestures is called a composite gesture. The primitive gestures constitute a basic vocabulary of gestures, i.e., each primitive gesture has a well defined meaning.

The primitive gestures can be concatenated. i.e., connected in a series of commands. Some gestures have easily recognized shapes, so that their beginning and endpoints can be detected in a composite gesture. However, most primitive gestures are difficult to isolate in a composite gesture. To deal with this, the primitive gestures are connected in series using a gestural operator called a "break" feature. The composite gesture is scanned to detect the break features or the beginnings and ends of such gestures connected without break feature. When found, the composite gesture is broken up into primitive gestur s and each gestur is interpreted and xecuted in th order in which

the individual gestures were drawn. The advantag of a distinct break feature is that new primitiv gestur s added to the user interface can be combined with existing gestures in any way needed by the user.

The preferred break feature is a sharp angle drawn between consecutive primitive gestures.

In the special case where the first primitive gesture in a composite gesture is a selection gesture, the break feature may be recognized as any departure from the selection gesture.

In accordance with the present invention, a graphic editing system is provided which comprises a data interface surface having identifiable coordinates, an application program including means for displaying data on said surface, means for interacting with said surface to control said application, to modify said data displayed on said surface and to generate a composite stroke on said surface, said composite stroke including a plurality of gestures, each representing a specified display operation on said surface, and means responsive to generating said composite stroke for executing each of said plurality of gestures.

In accordance with an aspect of the invention, a graphic editing system is provided which comprises a data interface surface having identifiable coordinates, an application program including means for displaying data on said surface, means for interacting with said surface to control said application, to modify said data displayed on said surface and to generate a composite stroke on said surface, said composite stroke including a plurality of gestures, each representing a specified operation, said composite stroke including at least one break feature for separating consecutive gestures of said plurality of gestures, and means responsive to generating said composite stroke for executing each of said plurality of gestures, said executing means including means for detecting said break features.

In accordance with another aspect of the invention, a graphic editing system is provided which comprises a data interface surface having identifiable coordinates, an application program including means for displaying data on said surface, means for interacting with said surface to control said application, to modify said data displayed on said surface and to generate a composite stroke on said surface, said composite stroke including first and second consecutive gestures, each representing a specified command for modifying said data displayed on said surface, said composite stroke including a means for separating said consecutive gestures of said plurality of gestures, and means responsive to generating said composite stroke for ex cuting each of said plurality of gestures, said executing means including means for detecting said separating means.

In accordanc with anoth r aspect of the invention, a graphic editing system is provided which comprises a data interface surface, an application program including means for displaying data on said surface, means for interacting with said surface to control said application, to modify said data displayed on said surface and to generate a composite stroke on said surface, said composite stroke including first and second commands, each having a defined shape, and means responsive to generating said composite stroke for detecting each shape of said first and second gestures.

In accordance with another aspect of the invention, a method for graphical editing is provided which includes a data interface surface and an application program including means for displaying data on said surface and means for interacting with said surface to control said application and to modify said data displayed on said surface. The method comprises the steps of generating a composite stroke on said surface including plurality of gestures, each representing a specified command for modifying said data displayed on said surface, and executing each of said plurality of gestures in response to generating said composite stroke.

The invention is illustrated by way of example and not limited by the figures of the accompanying drawings in which like references denote like or corresponding parts and in which:

Fig. 1 is a block diagram illustrating one form of touch based input apparatus in which the system according to the present invention can be employed;

Fig. 2 is an elevational view of a display screen with a loop gesture enclosing several objects;

Fig. 3 is a flowchart showing the sequence of operations for object manipulation;

Figs. 4a-d illustrate four primitive command gestures used as the basic tokens of gestural syntax;

Figs. 5a and 5b illustrate two different composite gestures consisting of two primitive gestures concatenated using a break feature;

Figs. 6a and 6b illustrate two additional primitive gestures;

Fig. 7 shows a composite gestural stroke consisting of the primitive gestures shown in Figs. 4a, 4d and 6b which are concatenated by incorporating the break feature in a preferred embodiment of the present invention;

Fig. 8 illustrates two primitive gestures concatenated by incorporating the break feature in another embodiment of the present invention; and

Figs. 9a and 9b illustrate two composite gestures consisting of two concat nat d primitive gestures.

Referring to Fig. 1, there is shown a block diagram of th touch based input system 10 including a CRT display 12. A touch s nsitive panel 14 is attached onto the surface of CRT display 12. Touch panel 14 is adapted to be touched by a user and such touch is detected by touch detection circuit 18. A detection signal corresponding to the position of the touched input point is output from touch detection circuit 18 in a controller 16.

The detected signal from touch detection circuit 18 is input to an X-Y detection circuit 20. X-Y detection circuit 20 processes the input signal and performs an arithmetic operation or the like. Thus, the X-Y detection circuit 20 detects the (x, y) coordinates of the input point touched on the surface of touch panel 14 and transmits such information to CPU 22. Touch panel detection circuit 18, X-Y detection circuit 20 and the CPU 22 combine to form controller 16. Controller 16 performs the control in accordance with the control program stored in program ROM 26 in memory 24. ROM section in memory 24 includes program ROM 26 in which the control program of CPU 22 is stored and pattern data area to store various kinds of character patterns or the like. RAM section 28 of memory 24 includes a RAM portion which is used as a work area of CPU 22 and a character position data area 30 to store display positions of character patterns and the like.

Specifically, in Fig. 2, touch panel 14 is an electronic input and display device such as an electronic sketch pad, liveboard or white board which employs a working surface and may employ a plurality of accessible functions 40 as is shown. The working surface is the upper area 36 of touch panel 14 and accessible functions 40 are displayed at the bottom area 38 of touch panel 14. These functions 40 may include new operation (draw mode), delete, move, shrink and so on which may be invoked by touching the respective display area. Alternatively, these functions can be accessed by a pie-menu. These functions however are optional in designation, their principal objects being to define operations which are inherent in the operation of the system. These functions may share the some of the same functions that are represented by many designated command gestures.

A suitable user manipulable control or interacting device such as a stylus or light pen or the like 42 is employed to draw input symbols, select functions or draw gestures representative of specified functions. Obviously, other variants known to persons skilled in the art may be employed.

In Fig. 2 the user has created objects 34 and has selected these objects by drawing a freeform selection stroke 32 around them.

A drawing editing application may incorporate the pr sent invention by making us of th following user object operations as illustrated in Fig. 3 wherein a prototypical control structure for a computer-assisted graphical editing tool incorporating the invention is shown. The sequence of operation is as follows.

When touch panel 14 is in DRAW mode, the system waits (step 50) for input on touch panel 14 from a user. The user can draw strokes on touch panel 14 to input data or enter a gesture command.

At step 62 the system determines if the input is a gesture by detecting some action taken on the part of the user. In the preferred embodiment, the user may inform the system of a gesture by pressing or exerting pressure on a button located on the stylus. However, there are other means available to signal the system that a gesture is being entered. If the input is not a gesture then the system performs other operations on desired data at step 58, and then the system returns to step 50 and waits for further input from the user.

If however a gesture stroke is detected, the stroke is scanned (step 64) and the presence of break features determined at step 65. If no break features are detected, the system will execute the primitive gesture operation (step 67). If at least one break feature is detected, then the system breaks the gesture into primitive gestures at step 66. Then the system interprets each primitive gesture as an operation, one after another, i.e., in the order in which they were drawn by the user. As each gesture is interpreted, the operation associated with each is executed at step 67. The system then returns to step 50 to wait for input.

Figs. 4a-d show four primitive gestures: *select* 70 (freeform loop around the data to be selected), *select list item* 72 (a bracket to the left of the list item to be selected), *delete* 74 (pigtail shape), and *move* 76 (a line beginning at a selection enclosure and ending at the new location) respectively. The user can combine any selection gesture with any one of the operation gestures. A break feature can be generally defined as any departure from the shape of the selection gesture. Thus, for example, as shown in Fig. 5a, the user can draw in a single stroke a selection gesture 70 and a move gesture 76 shown in Figs. 4a and 4d respectively or, as shown in Fig. 5b, select a listed item 72 and delete it 74.

Figs. 6a and 6b show two additional primitive gestures: *copy* 78 (a C at the end of the move line gesture) and *shrink* 80 (a spiral shape) respectively. These two additional gestures can be combined with the existing primitive gestures shown in Figs. 4a-d to allow a number of select-operate composite gestures and many more possible combinations can be formed by concatenating several operation gestures. For example, as is shown in Fig. 7, data (not shown) can be select d by

freeform selection gesture 70, moved by gesture 76 and then shrunk by gesture 80 by means of a single composite gesture. Whil each of th above composite gestures contains a selection gesture in order to provide an object upon which the operation command gestures operate, it is apparent that composite gestures need not necessarily employ such a selection gesture. For example, in Fig. 7, if the information was previously selected 70, then a composite gesture consisting of only move 76 and shrink 80 could be applied to the selection.

In Fig. 7, a sharp angle (A) is used as the break feature. The sharp angle is used as the preferred break feature since it is more natural and faster to draw. The sharp angle however may be disadvantageous because it prevents certain primitive gestures from being included, such as a scratchout gesture which has several sharp angles within it. Another preferred break feature is a large dot 82. See Fig. 8. This break feature does not have the restrictions of the sharp angle feature, but it is somewhat slower to perform.

It is noted that break features can be context independent or, to varying degrees, context dependent. A break feature is completely context independent if it can be recognized by the system in any composite input gesture without making any assumptions about the content of the gesture. For example, the "fat dot" feature would work as a context independent break feature over a set of primitive gestures that contained no "fat dots" themselves or any other kind of similar symbol that might be confused as a "fat dot."

On the other hand, an example of context dependence is an assumption that a selection gesture is always the first gesture in a composite gesture. This may avoid confusion between break feature and command gesture. If the system incorporates this assumption, then the break feature after a loop is any departure from the loop (sharp corner or no sharp corner). For example, the two gestures in each of Figs. 9a and 9b are recognized as a selection plus a move gesture.

Ideally, a context independent break feature is preferred. But the tradeoff of simplicity for this ideal may be speed and/or restrictions on the particular gesture set.

Claims

1. A graphic editing system comprising:
   (a) a data interface surface having identifiable coordinates;
   (b) an application program including means for displaying data on said surface;
   (c) means for interacting with said surface to control said application, to modify said data displayed on said surface and to g nrate a composite stroke on said surfac , said composite stroke including a plurality of gestures, each representing a specified display operation on said surface, and
   (d) means responsive to generating said composite stroke for executing each of said plurality of gestures.

2. A system according to claim 1, wherein: said composite stroke includes means for separating consecutive gestures of said plurality of gestures; and said executing means including means for detecting said break features.

3. The system according to claim 2 wherein said means for separating consecutive gestures of said plurality of gestures, comprises a break feature, said break feature preferably comprising (1) a sharp angle or (2) a dot, between consecutive gestures, or (3) a wedge connecting consecutive gestures.

4. The system according to claim 1, 2 or 3, wherein said executing means further includes means responsive to said detecting means for interpreting each of said plurality of gestures as an operation.

5. The system according to any of claims 1 to 4, wherein said executing means comprises means for executing said gestures of said plurality of gestures in a sequence defined by said application program.

6. The system according to any of the preceding claims further comprising means responsive to interacting of said first means with said surface for storing coordinates of said data displayed on said interface surface.

7. The system according to claim 6 further comprising means for retrieving said data from said storing means and displaying said data on said surface.

8. A graphic editing system comprising:
   (a) a data interface surface;
   (b) an application program including means for displaying data on said surface;
   (c) means for interacting with said surface to control said application, to modify said data displayed on said surface and to generate a composite stroke on said surface, said composite stroke including first and second commands, each having a defined shape; (d) means responsive to generating said composite stroke for det cting each shape of said first and second commands.

9. The system according to claim 8 further comprising means for executing said first command,said first command for xampl being for selecting specified data displayed on said surface.

10. A method for graphical editing including a data interface surface and an application program including means for displaying data on said surface, and means for interacting with said surface to control said application and to modify said data displayed on said surface, the method comprising the steps of;

generating a composite stroke on said surface including plurality of gestures, each representing a specified command for modifying said data displayed on said surface; and

executing each of said plurality of gestures in response to generating said composite stroke.

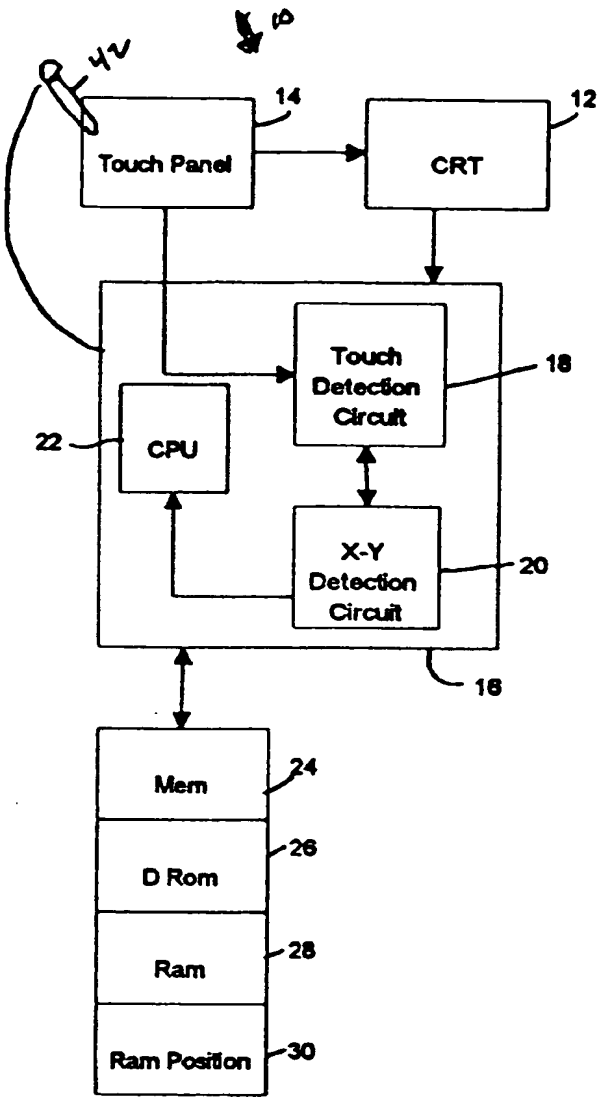
10
42
Touch Panel
14
CRT
12
Touch Detection Circuit
18
CPU
22
X-Y Detection Circuit
20
16
Mem
24
D Rom
26
Ram
28
Ram Position
30

Fig. 1

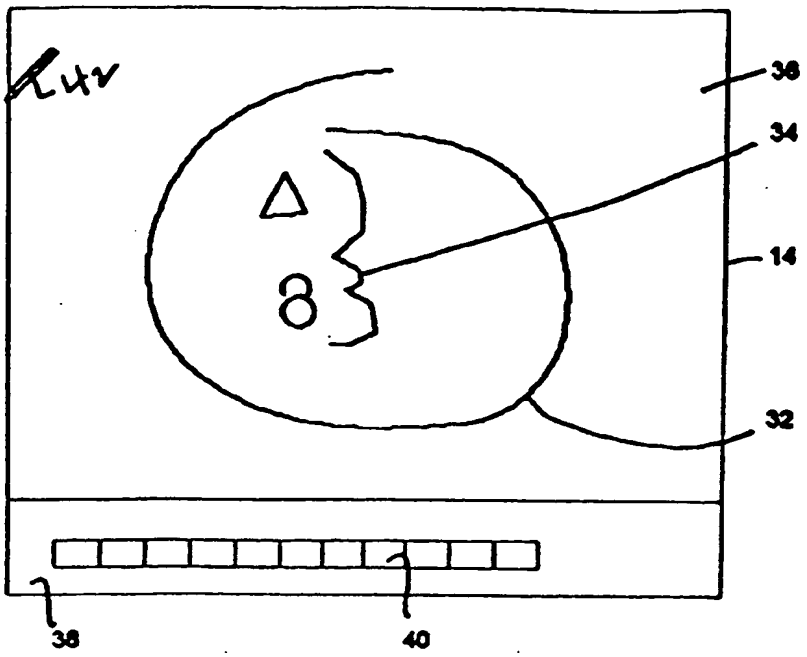

Fig. 2

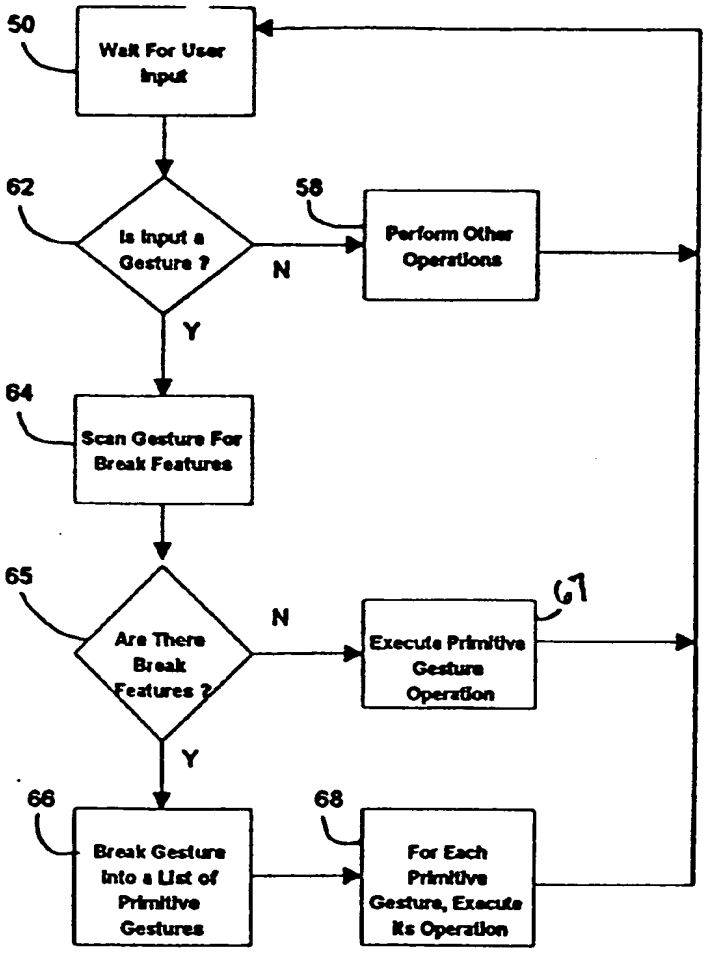
50
Wait For User Input
62
Is Input a Gesture ?
N
58
Perform Other Operations
Y
64
Scan Gesture For Break Features
65
Are There Break Features ?
N
67
Execute Primitive Gesture Operation
Y
66
Break Gesture Into a List of Primitive Gestures
68
For Each Primitive Gesture, Execute Its Operation

Fig. 3

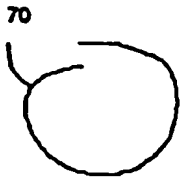

Fig 4a

Fig 4b

Fig 4c

Fig 4d

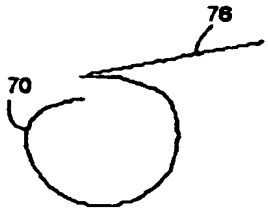

Fig 5a

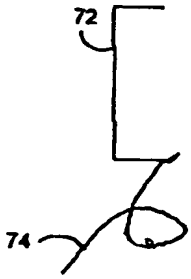

Fig 5b

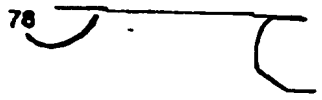

Fig 6a

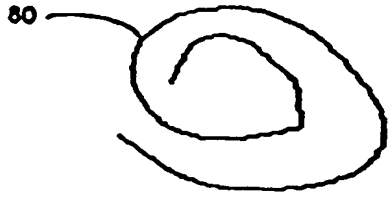

Fig 6b

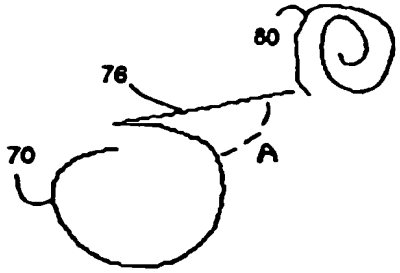

Fig 7

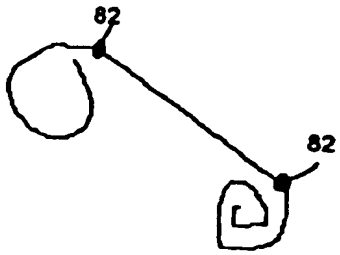

Fig 8

Fig 9a

Fig 9b

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 94 30 9234

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.6)
X	PROCEEDINGS OF THE ACM SYMPOSIUM ON USER INTERFACE SOFTWARE AND TECHNOLOGY, November 1991, S.C. USA pages 137 - 144, XP315074 KURTENBACH G. ET AL 'ISSUES IN COMBINING MARKING AND DIRECT MANIPULATION TECHNIQUES' * page 140, left column, line 26 - page 142, left column, line 47 * ---	1,5-10	G06F3/033
X	GRAPHICS INTERFACE 1991, June 1991, TORONTO,ON,CA pages 77 - 84, XP372056 HARDOCK G. 'Design Issues for Line-Driven Text Editing Annotation Systems' * page 78, right column, line 28 - line 38; figures 1,2 * -----	1,5-10	
			TECHNICAL FIELDS SEARCHED (Int.Cl.6)
			G06F

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	5 April 1995	Bailas, A

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)